Research Journal of Applied Sciences, Engineering and Technology 15(2): 57-67, 2018
DOI:10.19026/rjaset.15.5292
ISSN: 2040-7459; e-ISSN: 2040-7467

Submitted: August 8, 2017 Accepted: September 14, 2017 Published: February 15, 2018

Research Article

Arabic Sign Language Recognition Using Kinect Sensor

[1]Abdel-Gawad Abdel-Rabouh Abdel-Samie, [1]F.A. Elmisery, [1]Ayman M.Brisha and [2]Ahmed H. Khalil
[1]Department of Electronics Technology, Faculty of Industrial Education,
Beni-Suef University, Beni-Suef,
[2]Department of Electronics and Communications, Faculty of Engineering, Cairo University, Giza, Egypt

Abstract: This study introduces a Real Time System for automatic Arabic sign language recognition system based on Dynamic time warping matching algorithm. The communication between human and machines or between people could done using gestures called sign language. The aim of the sign language recognition is to give an exact and convenient mechanism to transcribe sign gestures into meaningful text or speech so that communication between deaf and hearing society can easily be made. In this study we introduce a translator based on Dynamic Time Warping, where each signed word is coordinating and matching among database, then display the text and the corresponding pronunciation of the income sign. We using the Microsoft's Kinect sensor to catch the sign. We have built our data using a large set of samples for a dictionary of 30 isolated words homemade signs from the Standard Arabic sign language. The system operates in different modes including online, signer-dependent and signer-independent modes. The presented system allows the signer to do signs freely and naturally. Experimental results using real Arabic sign language data collected show that the presented system has higher recognition rate compared with others for all modes. For signer-dependent online case, the system achieves recognition rate of 97.58%. On the other hand, for signer-independent online case, the system achieves a recognition rate of 95.25%.

Keywords: Arabic sign language recognition ArSL, DTWA, kinect, microsoft visual studio, real-time

INTRODUCTION

Sign Language (SL) as a kind of gestures is one of the most natural ways of communication for most people in deaf community. Recently sign language has been broadly planned to set up using many input devices, such as web camera, data glove, etc. (Poddar *et al*., 2015). The aim of the sign language recognition is to give an exact and convenient mechanism to transcribe sign gestures into meaningful text or speech so that communication between deaf and hearing society can easily be made (AL-Rousan *et al*., 2009). Though data glove-established SL translator works better for huge numbers of signs, but the data glove is too costly. On the other hand, vision-based approach is most suitable, user-friendly and affordable. So, it is widely used. With a web camera SL translator is correct and free body movements. However, it is tough for backgrounds and brightness. On the other hand, SL translation using the depth images value acquired by Microsoft Kinect 360^{TM} is an RGB-D sensor providing synchronized colour and depth images (Kyatanavar and Futane, 2012). It was initially used as an input device by Microsoft for the Xbox game console (Han *et al*., 2013). The learning and recognition methods used in earlier studies to automatically recognize SLR include neural networks, rule-based matching, hidden Markov models and Dynamic time warping matching algorithm.

In this study we built a Real Time System for automatic Arabic sign language recognition system based on Dynamic time warping matching algorithm to find the Arabic sign language to help communication with the deaf or teaching the Arabic sign language to any user. Therefore, the word signer was using in this study to include the deaf and others, using the data provided by the Microsoft Kinect 360^{TM}camera. We used the Dynamic time warping matching algorithm to compare the income sign with the references signs. The output of the translator will give the best match of each sign, so that a computer will output the corresponding word to a sign executed by a signer in front of a Microsoft Kinect, as well as the sound of this word.

LITERATURE REVIEW

Sign Language Recognition (SLR) could categorized into isolated SLR and continuous SLR and each could further classified into signer-

Corresponding Author: Abdel-Gawad Abdel-Rabouh Abdel-Samie, Department of Electronics Technology, Faculty of Industrial Education, Beni-Suef University, Beni-Suef, Egypt, Tel.: +00201064021965

dependent and signer-independent according to the sensitivity to the signer. Also one may classify SLR systems as either glove-based, if the system relies on electromechanical devices for data collection, or none glove-based, if bare hands used. Many of review process on Human Gesture Recognition had presented before in Pavlovic *et al*. (1997), We and Huang (1999) and Gavrila (1999). They are mostly used 2D information and only minority of them worked with depth data (3D). Starner *et al*. (1997) used a view-based approach with a single camera to extract 2D features as the Enter of HMM for continuous American Sign Language. They got Word accuracy of 92% or in recognizing the sentences with 40 different signs (Starner *et al*., 1997). In Youssif *et al*. (2011) introduces an automatic Arabic sign language recognition system based on the (HMM). A large set of samples had used to recognize 20 isolated words from the Standard Arabic sign language. The proposed system is signer-independent. Experiments are conducting using real Arabic sign language videos taken for deaf people in different clothes and with different skin colours. This system achieves an overall recognition rate reaching up to 82.22% (Youssif *et al*., 2011). In Poddar *et al*. (2015) used a webcam to recognize the hand positions and sign made using contour recognition and outputs the SL in PC to the gesture made (Poddar *et al*., 2015). In Vogler *et al*. (2000) introduced the parallel Hidden Markov Model-based method. They used 3D data as the Enter of the recognition magnetic tracking system such as the Ascenion Technologies Motion Star system. They showed how to apply this framework in practice with successful results using a 22-sign-vocabulary. The reported best accuracy is 95.83% (Vogler *et al*., 2000). In Raheja *et al*. (2015) a hand gesture recognition method using the Microsoft Kinect had proposed, which operates robustly in uncontrolled environments and is insensitive to hand variations and distortions. This demonstrates used of two different learning techniques, dynamic time warping and hidden Markov model and compares them for real-time implementations. The recognition success rate was over 90% (Raheja *et al*., 2015). In Hee-Deok Yang. Used 3D depth information from hand motions, generated from Microsoft's Kinect sensor and apply a hierarchical Conditional Random Field that recognizes hand signs from the hand motions. The method used a hierarchical Conditional Random Field to detect candidate segments of signs using hand motions and then a Boost Map embedding method to verify the hand shapes of the segmented signs. Experiments demonstrated that the method could recognize signs from signed sentence data at a rate of 90.4% (Yang, 2014). In Daniel M. the Microsoft Kinect is proposed to solve the problem of sign language translation. Using the tracking ability of this RGB-D camera, a meaningful 8-dimensional descriptor for every frame is introducing here. In addition, an efficient Nearest Neighbour dynamic time warping and Nearest Group dynamic time warping are developing for fast comparison between signs. With the proposed descriptors and classifiers combined with using the Microsoft Kinect, for a dictionary of 14 homemade signs, the introduced system achieves an accuracy of 95.238% (Capilla, 2012).

Arabic sign language are still in their development stages. According to the unified Arabic sign language approved by the Arab League of States (Arab League of States, 2001) the ArSL dictionary has more than 1400 signs. Figure 1 shows samples of Arabic sign language ("Pray**يصلى**","Allah **الله**"," A guest**ضيف**"," Referee**حكم**"," There**هناك**"," Get in **تفضل** "," Phone**هاتف** "," welcome**أهلا**"," I**أنا**","peace upon you "," Father أب "," Trousers").

Fig. 1: Sample sign of Arabic sign language (ArSL)

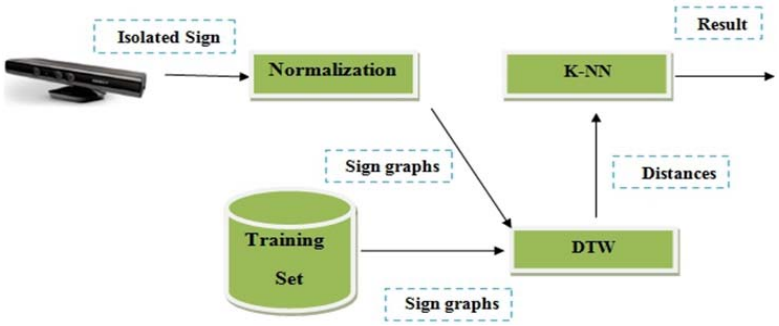


Fig. 2: Overview of the system

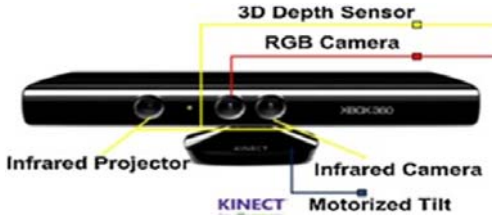


Fig. 3: Kinect device

MATERIALS AND METHODS

Over view of our system shown in Fig. 2. In this figure the signer is standing in front of the kinect sensor to make the sign. The output from the kinect is normalized, then compared with the training set. Finally display the best match for the income sign. We took care for our system to an interactive user interface so that the user will be able to run the application without any previous knowledge. The system works on real-time. Allow the user to auto-train the dictionary by adding new words. Get the text and sound for each word recognized in the output.

The Kinect XBOX 360^{TM}: The Kinect Sensor developed by Microsoft and Prime Sense (Fig. 3). It is a hardware device used to control the Microsoft Kinect game console without any kind of controller that the user has to hold or wear. It supports the Depth image including player index.RGB image. Tilt (Get and Set). Microphone Array. Skeleton Tracking.

Human skeleton can divided into two parts as upper body and lower body. Kinect for Windows can track twenty joints of human body, half of them belonging to the upper body while the other half belonging to the lower body. Upper body joints consist of right hand, right wrist, right elbow, right shoulder, head and center of shoulders, left shoulder, left elbow, left wrist and left hand. Lower body joints consist of rightfoot, right ankle, right knee, right hip, spine, center of hips, left hip, left knee, left ankle and left foot. These joints shown in Fig. 4. After carefully studying the signs of the proposed default dictionary for our system, only 12 joints out of the 20 resulted to significant for description the sign: both hands, both elbows, both wrists and both shoulder. These are the head, shoulder center, spine and hip centers joints. By doing so, the list of tracked joints at every frame reduced from 20 to 12 shown in Fig. 5. To deal with the Kinect sensors the Open NI Frame work was used. This is an open source package by Prime Sense (Hussein *et al*., 2014).

To start using the kinect, there is a start-upposition for the user, this start-up position as shown Fig. 6. Once the calibration done, Kinect tracks the joints and limbs position.

Data normalization for user's position: Data normalization needed because of the distances between one joint and another one can drastically vary depending on position of the signer. The signer can exist at different positions of the room and so the data must store according to that position. In Fig. 7, a slight variation in-depth can cause a much variation of the X and Y values. In our application we fixed the position of the signer to overcome this problem.

Data normalization for user's size: Given a sign, its description must the same no matter if the user is tall or short and the translator must able to output the right word in every case. There is no way to add the samples for all the possible user's sizes to the dictionary. Otherwise the classification process will become slower and its accuracy is few. The user's size problem as shown in Fig. 8a.

The distance from one joint to another changes significantly depending on the user's size (the distances for the users in the middle are smaller than the distances for the users at the sides). After normalization of the user's position, every joint expressed by its relative distance d to the spine joint. The presented shown in Fig. 8b consists of normalizing all the relative distances d by the factor that defined by the distance between the HEAD and the SPINE joints (d_{HS}). This factor tells about the size of the user and all the distances D can normalized according to this value. Given the set of distances $D = \{d_{RS}, d_{LS}, d_{RE}, d_{LE}, d_{RH}, d_{LH}\}$ the normalized set of distances D norm obtained as follows:

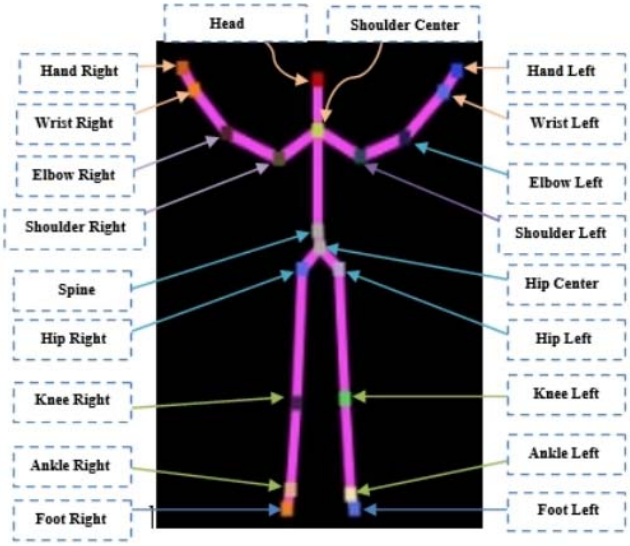


Fig. 4: Skeleton tracking

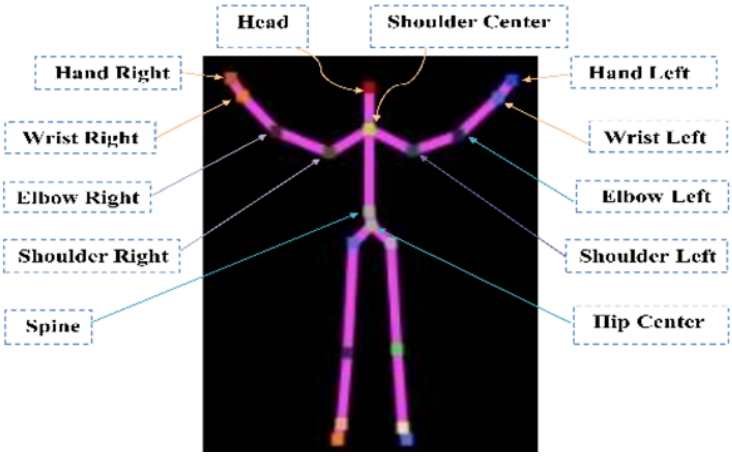


Fig. 5: Used joints

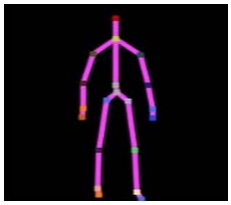


Fig. 6: Calibration position

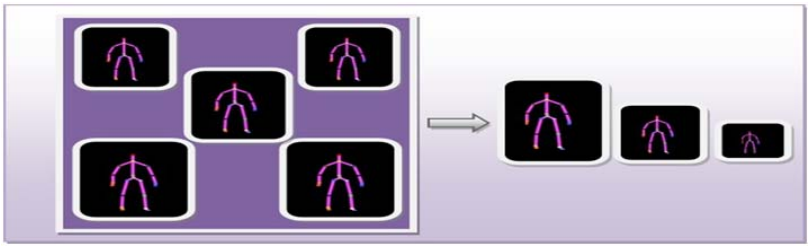
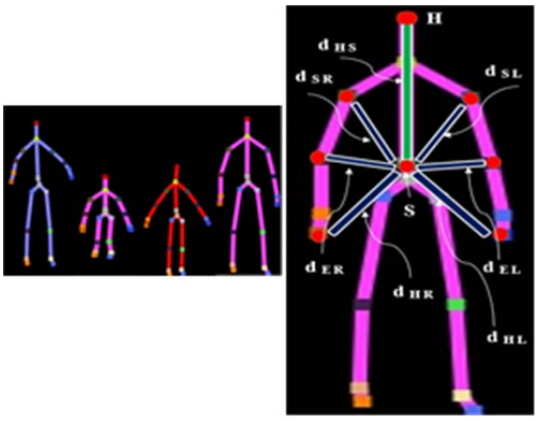


Fig. 7: Normalization required for the position of the user



(a) Different user sizes (b) Set of distances D

Fig. 8: Normalization required for the user sizes

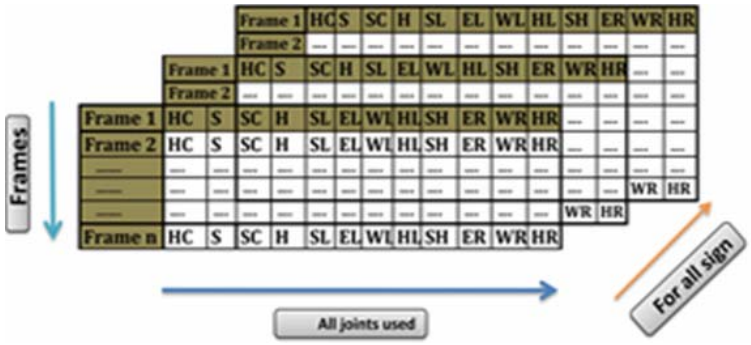


Fig. 9: Sign descriptor for every frame and joint

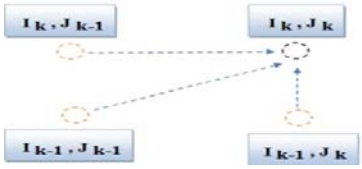


Fig. 10: Predecessor nodes used in Bellman's principle

$$\sum_{i=1}^{n} D_{norm}(i) = \frac{D_{(i)}}{d_{HS}} \quad (1)$$

where n is the number of distances from D and d_{HS} is the HEAD-SPINE distance (the green segment from image in Fig. 8b. There is no need to normalize the angles since they are expressing the direction and the direction remains the same after the normalization.

Training: Once the Joints of interest the data obtained and normalized, the next step is building a descriptor for each sign. The descriptor must able to describe a sign in a way that this descriptor will be unique and sufficiently different from the other descriptors of the dictionary as shown in Fig. 9.

Dynamic time warping: After normalization, a sign represented as a set of joint paths each of which keeps track of trajectory of a joint during production of that sign. Because joint paths for a sign obtained from the frames of a single skeleton, the number of elements in a joint path sequence (frames) is the same for all joint paths belonging to a single production of a sign. Though, it might have different values for distinct production of the same signs by the same signer. As a result, training and test data extracted from skeletons will possibly have some non-linear variations in time. Therefore, we use dynamic time warping which is an adequate technique for classifying trajectories of joints in the way that it can rate joint paths independent of non-linear variations in the characteristics of the data. Since all the elements ordered in time, the set of predecessor nodes are to the left and bottom of a current node shown in Fig. 10.

The least cost path is optimal alignment between two sequences. One way to find the least cost path is to test every possible path from the left-bottom corner to right-top corner. A local distance d (i, j) between any two feature vectors r (i) and t (j) can calculated using Manhattan, Euclidean, or sum squared differences distances. We used the Euclidean distance to calculate the local distance which give us the best results. In dynamic time warping the global distance D (i, j) of any

Table 1: Dictionary of default signs for training and testing data in the system

No	Class (Sign)	Number of test samples
1	(n) Family	40
2	(n) Girl	40
3	(n) Baby	40
4	(n) Father	40
5	(n) Allah	40
6	(n) Home	40
7	(n) Phone	40
8	(n) Referee	40
9	(n) Cairo	40
10	(n) Car	40
11	(adv)	40
12	(v) Sleep	40
13	(v) Drink	40
14	(v) Sniff	40
15	(v) Love	40
16	Beside	40
17	There	40
18	(n) A guest	40
19	Peace upon you	40
20	(v) Get in	40
21	welcome	40
22	Thanks	40
23	Hello	40
24	(n) heart	40
25	(v) Pray	40
26	(n) Television	40
27	(n) Trousers	40
28	O Allah	40
29	Here	40
30	(n)Soldier	40
Total		1200

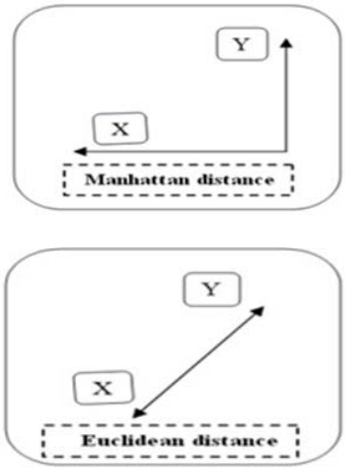


Fig. 11: Difference between Euclidean and Manhattan Distance

two feature vectors r (i) and t (j) computed recursively by adding its local distance d (i, j) to the evaluated global distance for the best predecessor. The best predecessor is the one that gives the least global distance D (i, j) (see Eq.2) at row i and column j with m≤ i and k≤ j:

$$D(i,j) = \min[(m,k)] + d(i,j) \quad (2)$$

The total cost D of the mapping between r and t with respect to a distance function d (i, j), defined as the sum of all distances between the mapped sequence elements) (see Eq.3):

$$D = \sum_{k=0}^{f} d(ik, jk), \quad (3)$$

where, d (i, j) measures the distance between elementsr (i) and t (j) (Celebi *et al*., 2013).

DynamicTime Warping guarantees to find an ideal warping path which has the lowest total cost compared to all possible warping paths even if there exists more than one ideal path. We have applied the training method of dynamic time warping as described above used as the recognition algorithms for comparison in Arabic sign language recognition system using Kinect Sensor. By using the data default training set has 30 Arabic signs from the dictionary of words listed in Table 1 and steps of dynamic time warping Algorithm. We compute the min distance optimal alignment between two sequences. The recognized sign obtained using the formula by Find distance matrix (d) between the Feature Vectors using Euclidian distance and Manhattan distance (Akila and Chandra, 2013). After the comparison process completed, the best match obtained and the word is taking out corresponding to the sign made by the signer person see Eq. (4) and (5).

Euclidean distance: Thakurand Sahayam (2013) it is the most widely used distance measure of all available. In the Eq.4, the data in vector x and y are subtracted directly from each other. The Euclidean Distance (ED) between two-time series $\{X_i\}$ and $\{Y_i\}$ defined as:

$$ED = \sqrt{\sum_{i=1}^{n}(Xi - Yi)^2} \quad (4)$$

While ED is easy to define, it performs poorly as a similarity score. Since it aligns each point on one time series with the each point on another, ED is very sensitive to distortions in the time domain.

Manhattan distance: It is also known as City Block or Taxi Cab distance (Gene Expression Data Analysis Suite, gedas.bizhat.com/ dist.html). It is closely related to the Euclidean distance. The Euclidean distance corresponds to the length of the shortest path between two points, the city-block distance is the sum of distances along each dimension shown in Eq. (5). This is equal to the distance a traveler would have to walk between two points in a city. The Manhattan distance cannot move with the points diagonally, it has to move horizontally and vertically which shown in Fig. 11. The city-block distance is a metric, as it satisfies the triangle inequality. As for the Euclidean distance, the expression data subtracted directly from each other and thence must make sure that they are properly normalized:

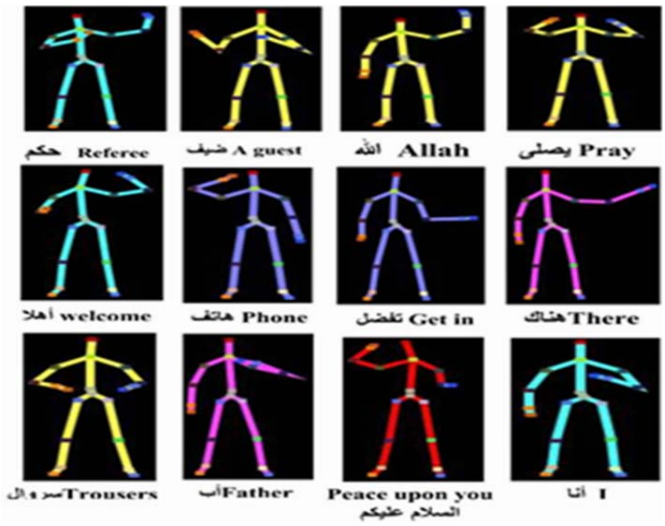


Fig. 12: Sample sign of Arabic sign language (ArSL) live taken by the trainee in front of Microsoft Kinect

Table 2: Sample of database for each sign (frames and joints)

Name & joint	Family	Girl	Baby	Father	Allah	Home
1	-0.369033	-0.6258	0.245861	-0.6674	-0.64833	-0.08802
2	-0.3537489	-1.5983	-0.60324	-1.57189	-1.74046	0.010811
3	-0.4786952	-0.6319	0.017029	-0.66175	-0.6361	-0.19284
4	-0.407177	-1.3537	-0.57056	-1.35618	-1.48573	-0.12583
5	-0.7553901	-0.6174	-0.66947	-0.62795	-0.644	-0.50729
6	-0.6355067	-0.7874	-0.47252	-0.6914	-0.81115	-0.53575
7	0.68846208	0.84152	0.658489	0.722837	0.906209	0.536076
8	-0.6307333	-0.037	-0.54469	-0.51766	0.363235	-0.55901
9	0.24274488	0.55668	0.047919	0.149494	0.814986	-0.04483
10	-0.3273778	0.40264	-0.65518	-0.26278	0.87974	-0.12568
11	0.24626529	0.48191	-0.1556	-0.04122	0.760117	0.053854
12	-0.2911876	0.64432	-0.69202	-0.17782	1.059898	0.004608

$$MD = \sum_{i=1}^{n} |Xi - Yi| \quad (5)$$

Classifier: In the classification phase we used the Nearest Neighbour to classify the output from the dynamic time warping. Given a sample test, it matched with most similar group of signs samples from the dictionary. The most similar group is the one with the smallest mean similarity.

ArSL database collection: Because there are no common databases available for Arabic sign language recognition. Therefore, we had to build our own database with reasonable size. As depicted in Fig. 1. The Kinect sensor used to acquire the signs from the signers in a frames format. The signers will use bare hands, i.e., no need for any gloves. The signers must use the unified version of the Arabic sign language which approved by the Arab League of States (Arab League of States, 2001).

In recoding sign, there is no limitation on the lighting or the background of the scene, neither does the clothing of the person. The signer starts from silence, does the required sign and ends in silence. In this phase, we recorded the sign at the rate of 32 frames per second. Image part of the sign considered, ignoring the audio part of the image. Our dataset consists of Arabic Sign Language signs captured using Kinect sensor. Sign data stored as frame-by-frame skeletons of a signer. Open NI framework used for skeleton tracking. Therefore, movements of the joints in 3D space can extracted easily from the set.

In our study, joint paths of the hands and the elbows extracted from skeleton information to generate sign graphs. The default training set has a total of 120 different samples from Arabic sign language, which is the result after adding four different samples for each sign. Our dataset has 30 signs shown in Table 1. All these training samples belongs to the same user and executed at the same position. Some of default training set by the trainee's in front of a Microsoft Kinect shown in Fig. 12. For each sign 32 frames and per frame 12 joints. These joints describe the sing coordinates within the program for all sign used in the training. Table 2 show some of these coordinates then saved in a file inside the program.

RESULTS AND DISCUSSION

In this study we used Microsoft Visual Studio(c#) to write the programs for our system, we employed the

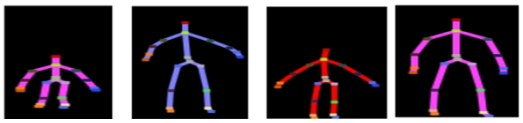


Fig. 13: Some different users used to test the system

Table 3: Recognition rates of the ArSL system for the test data

Mode & Number & Percentage	Number of correction	Percentage
Signer-dependent	1171 Sign	97.58%
Signer-independent	1143 Sign	95.25%

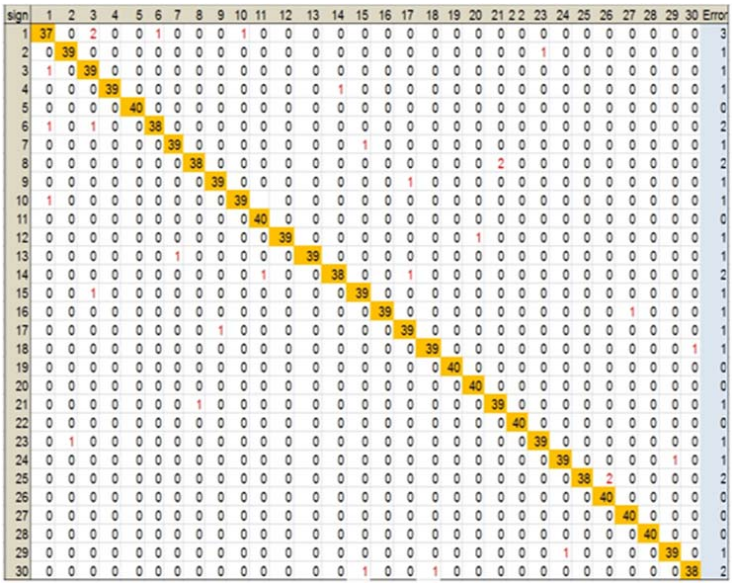
Microsoft Kinect as depth sensor, using the Open NI APIs to interface it and the NITE framework for depth image analysis and control skeleton extraction.

To evaluate our system some experiments carried out to evaluate performance of the presented system. A set of test samples collected. This set has signs done by 10 different signer (four women and six men). Some of different signer used to test the system as show in Fig. 13. The signers of different age, physical properties and different times. For every signer 4 different samples for each sign added to the set of test samples. This results in a total of 1200 testing samples that will be using to find the accuracy of the system.

We have conducted several experiments to evaluate our Arabic sign language recognition system. The signers perform signs while the system is ready to recognize the signs immediately through outputting the corresponding signs, text and voice. The system was able to recognize the income signs in real-time after processing the data.

In the first experiment was for signer-dependent evaluation using the training data collected. Depending on the database set. The overall system performance was 97.58% show in Table 3, which is reasonably high. The number of misclassifications is 29 out of 1200 signs which correspond to 2.42% error rate. The detailed per-sign misclassifications shown in Fig. 14.

The more appropriate indicative way of measuring the system performance is to test the system using different set from signer. In the second experiment was for signer-independent online evaluation was presenting a total of 1200 sign distributed among the signs shown in Table 1. The recognition accuracy went down a little. However, the system has shown excellent performance was 95.25% show in Table 3 with a low error rate of 4.75% corresponding to others. The resulted number of



sign	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	Error
1	37	0	2	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3
2	0	39	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
3	1	0	39	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
4	0	0	0	39	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
5	0	0	0	0	40	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	1	0	0	38	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2
7	0	0	0	0	0	0	39	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
8	0	0	0	0	0	0	0	38	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	2
9	0	0	0	0	0	0	0	0	39	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1
10	1	0	0	0	0	0	0	0	0	39		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
11	0	0	0	0	0	0	0	0	0	0	40	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	39	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1
13	0	0	0	0	0	0	1	0	0	0	0	0	39	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
14	0	0	0	0	0	0	0	0	0	1	0	0	0	38	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	2
15	0	0	1	0	0	0	0	0	0	0	0	0	0	0	39	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	39	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
17	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	39	0	0	0	0	0	0	0	0	0	0	0	0	0	1
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	39	0	0	0	0	0	0	0	0	0	0	0	1	1
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	40	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	40	0	0	0	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	39	0	0	0	0	0	0	0	0	0	1
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	40	0	0	0	0	0	0	0	0	0
23	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	39	0	0	0	0	0	0	0	1
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	39	0	0	0	0	1	0	1
25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	38	2	0	0	0	0	2
26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	40	0	0	0	0	0
27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	40	0	0	0	0
28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	40	0	0	0
29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	39	0	1
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	38	2

Fig. 14: Confusion matrix for test data in signer-dependent/online mod

Table 4: Comparison with similar online, signer-dependent and signer-independent systems or something else

No	Method	Recognation rate
1	Hidden Markov Models (HMM) (AL-Rousan *et al*., 2009).	93.8% dep and 90.6% indep case
2	NNDTW and NGDT (Capilla, 2012)	95.24%
3	Hidden Markov Models (HMM) (Mustafa and Demopoulos, 2014)	90%
4	Nearest Neighbours DTW (I ¸Siklig˙il, 2014)	91.0%dep and 59.3%indep case
5	Random Decision Forest (RDF) (Dong *et al*., 2015)	92%
6	Hierarchical Conditional Random Field (CRF) (Yang, 2014)	90.4%
7	(HMM) and DTW (Raheja *et al*., 2015)	90%
8	Dynamic gesture recognition (Chen *et al*., 2015)	95.42%
9	Dynamic time warping NN classifier (Ribó *et al*., 2016)	68.0% and 68.4%
10	Hidden Markov models (HMM) (Maruvada, 2017)	63.6% and 86.8%
11	Presented system with DTW and kinect sensor	97.58% dep and 95.25% indep case

misclassifications is 57 out of 1200 signs. The detailed per-sign misclassifications shown in Fig. 15.

We compared our results with previously published results in the field of Arabic sign language and others. The results shown in Table 4. It is noticeable that our system performs better and the presented system has shown excellent performance.

Real time of Arabic sign language recognition result figures testing in the algorithm: In Fig. 16a and 16b shown the signer is standing in front of the kinect sensor to make the sign and then the Skelton for this sign catched. After that the output for the corresponding word of the pervious sign displayed (text and voice) on the screen. So, the user will understand the sign.

Sign	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	Error
1	32	0	2	0	0	4	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	8
2	0	38	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	2
3	2	0	38	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2
4	0	1	0	37	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3
5	0	0	0	0	40	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	4	0	2	0	0	34	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6
7	0	0	0	0	0	0	36	0	0	0	0	0	0	0	1	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	4
8	0	0	0	0	0	0	0	37	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	3
9	0	0	0	0	0	0	0	0	39	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1
10	1	0	0	0	0	0	0	0	0	39	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
11	0	0	0	0	0	0	0	0	0	0	40	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	38	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	2
13	0	0	0	0	0	0	1	0	0	0	0	0	38	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	2
14	0	2	0	2	0	0	0	0	0	0	2	0	0	34	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	6
15	0	0	1	0	0	0	0	0	0	0	0	0	0	0	39	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	39	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1
17	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	38	0	0	0	0	0	0	0	0	0	0	0	0	0	2
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	39	0	0	0	0	0	0	0	0	0	0	0	1	1
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	40	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	40	0	0	0	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	39	0	0	0	0	0	0	0	0	0	1
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	40	0	0	0	0	0	0	0	0	0
23	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	38	0	0	0	0	0	0	0	2
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	38	0	0	0	0	2	0	2
25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	38	2	0	0	0	0	1
26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	40	0	0	0	0	0
27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	40	0	0	0	0
28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	40	0	0	0
29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	39	0	1
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	2	0	0	0	0	0	0	0	0	0	0	0	35	5

Fig. 15: Confusion matrix for test data in signer-independent/online mode

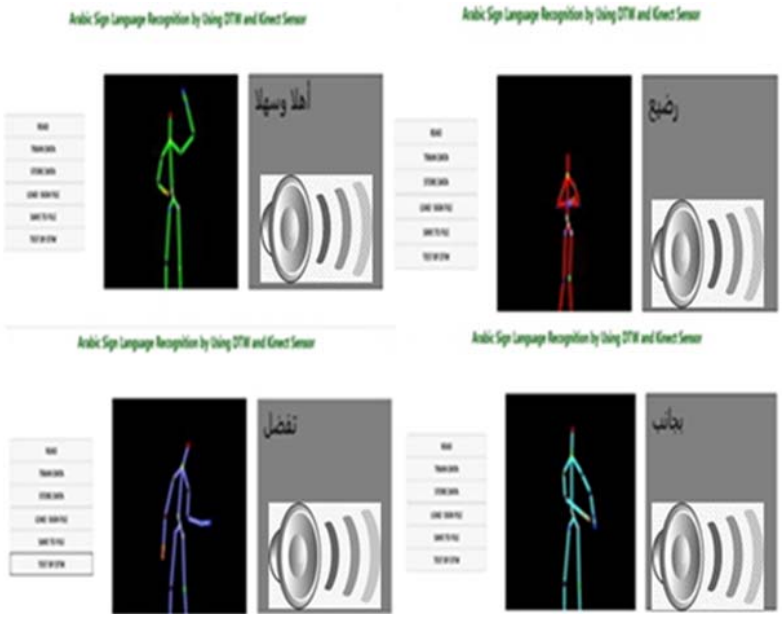


Fig. 16a: Some of output ArSL of our system by kinect on PC

Fig. 16b: Some of output ArSL of our presented system by kinect

CONCLUSION

In this study we introduce a Real Time System for automatic Arabic sign language recognition system based on Dynamic Time Warping matching algorithm by using Kinect Sensor. The system does not use any type of data/power gloves. A large set of samples had used to recognize for a dictionary of 30 isolated words homemade signs from the Standard Arabic sign language. The system operates in different modes including online, signer-dependent and signer-independent modes. We used the Dynamic Time Warping matching algorithm for comparing between signs. Experimental results show that the presented system has high recognition rate for all modes. For signer-dependent, the system achieves a recognition rate of 97.58% and 2.42% error rate. On the other hand, for signer-independent, the system achieves a recognition rate of 95.25% and 4.75% error rate.

REFERENCES

Akila, A. and E. Chandra, 2013. Slope finder - a distance measure for DTW based isolated word speech recognition. Int. J. Eng. Comput. Sci., 2(12): 3411-3417.

AL-Rousan, M., K. Assaleh and A. Tala'a, 2009. Video-based signer-independent Arabic sign language recognition using hidden Markov models. Appl. Soft Comput., 9(3): 990-992.

Arab League of States, 2001. The Arabic Dictionary of the Deaf. The Arab Sign Language Dictionary. Arab Organization for Education, Culture and Science in Tunis, Department of Social Development of the League of Arab States and the Arab Union of Deaf Welfare Organizations.

Capilla, D.M., 2012. Sign Language Translator Using Microsoft Kinect XBOX 360TM. In: Qi, H. and F. Meriaudeaum (Eds.), 4: 2-5. Retrieved from: https://pdfs.semanticscholar.org/165a/1a7c529f51b91ae587496d41603e560d1fe9.pdf.

Celebi, S., A.S. Aydin, T.T. Temiz and T. Arici, 2013. Gesture recognition using skeleton data with weighted dynamic time warping. Proceeding of the International Conference on Computer Vision Theory and Applications (VISIGRAPP, 2013)1: 2-2.

Chen, Y., B. Luo, Y.L. Chen, G. Liang and X. Wu, 2015. A real-time dynamic hand gesture recognition system using kinect sensor. Proceeding of the IEEE Conference on Robotics and Biomimetics. Zhuhai, China, 4: 6-9.

Dong, C., M.C. Leu and Z. Yin, 2015. American sign language alphabet recognition using Microsoft Kinect. Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). Boston, MA, USA.

Gavrila, D.M., 1999. The visual analysis of human movement: A survey. Comput. Vis. Image Und., 73(1): 82-98.

Han, J., L. Shao, D. Xu and J. Shotton, 2013. Enhanced computer vision with Microsoft Kinect sensor: A review. IEEE T. Cybernetics, 43(5): 1318-1334.

Hussein, M.A., A.S. Ali, F.A. Elmisery and R. Mostafa, 2014. Motion control of robot by using kinect sensor. Res. J. Appl. Sci. Eng. Technol., 8(11): 1384-1388.

I ¸Siklig˙il, E., 2014. A method for isolated sign recognition with kinect. M.Sc. Thesis, Department of Computer Engineering, Middle East Technical University.

Kyatanavar, R.D. and P.R. Futane, 2012. Comparative study of sign language recognition systems. Int. J. Sci. Res. Publ., 2(6): 1-3.

Maruvada, S., 2017. 3-D hand gesture recognition with different temporal behaviors using HMM and kinect. M.Sc. Thesis, University of Magdeburg.

Mustafa, E. and K. Demopoulos, 2014. Sign language recognition using Kinect. Proceeding of the 9th South East European Doctoral Student Conference. Thessaloniki, Greece, pp: 1-15.

Pavlovic, V.I., R. Sharma and T.S. Huang, 1997. Visual interpretation of hand gestures for human-computer interaction: A review. IEEE T. Pattern Anal., 19(7): 677-695.

Poddar, N., S. Rao, S. Sawant, V. Somavanshi and S. Chandak, 2015. Study of sign language translation using gesture recognition. Int. J. Adv. Res. Comput. Commun. Eng., 4(2): 264-267.

Raheja, J.L., M. Minhas, D. Prashanth, T. Shah, A. Chaudhary, 2015. Robust gesture recognition using Kinect: A comparison between DTW and HMM. Optik Int. J. Light Electr. Optics, 126(11-12): 1098-1104.

Ribó, A., D. Warchoł and M. Oszust 2016. An approach to gesture recognition with skeletal data using dynamic time warping and nearest neighbour classifier. I.J. Intell. Syst. Appl., 6: 1-8.

Starner, T., J. Weaver and A. Pentland, 1997. A wearable computer-based American sign language recogniser. Pers. Technol.,1(4): 241-250.

Thakur, A.S. and N. Sahayam, 2013. Speech recognition using euclidean distance. Int. J. Emerg. Technol. Adv. Eng. 3(3): 587-590.

Vogler, C., H. Sun and D. Metaxas, 2000. A framework for motion recognition with applications to American sign language and gait recognition. Proceeding of the IEEE Workshop on Human Motion, pp: 33-38.

We, Y. and T.S. Huang, 1999. Vision-Based Gesture Recognition: A Review. In: Braffort, A., R. Gherbi, S. Gibet, D. Teil and J. Richardson (Eds.), Gesture-Based Communication in Human-Computer Interaction. GW 1999. Lecture Notes in Computer Science (Lecture Notes in Artificial Intelligence), Vol. 1739. Springer, Berlin, Heidelberg.

Yang, H.D., 2014. Sign language recognition with the kinect sensor based on conditional random fields. Sensors, 15(1): 135-147.

Youssif, A.A.A., A.E. Aboutabl and H.H. Ali, 2011. Arabic sign language (ArSL) recognition system using HMM. Int. J. Adv. Comput. Sci. Appl., 2(11): 45-51.